

US007797134B2

(12) **United States Patent**  
**Barr et al.**

(10) **Patent No.:** **US 7,797,134 B2**  
(45) **Date of Patent:** **Sep. 14, 2010**

(54) **SYSTEM AND METHOD FOR TESTING A MEMORY WITH AN EXPANSION CARD USING DMA**

(75) Inventors: **Andrew H. Barr**, Roseville, CA (US);  
**Ken G. Pomaranski**, Roseville, CA (US);  
**Dale J. Shidla**, Roseville, CA (US)

(73) Assignee: **Hewlett-Packard Development Company, L.P.**, Houston, TX (US)

(\*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 844 days.

5,423,029	A *	6/1995	Schieve	714/42
5,446,741	A	8/1995	Boldt et al.	
5,551,006	A *	8/1996	Kulkarni	711/146
5,619,513	A	4/1997	Shaffer et al.	
5,668,815	A	9/1997	Gittinger et al.	
5,890,012	A *	3/1999	Poisner	710/22
6,002,868	A *	12/1999	Jenkins et al.	717/105
6,263,452	B1 *	7/2001	Jewett et al.	714/9
6,421,798	B1 *	7/2002	Lin et al.	714/718
6,480,982	B1	11/2002	Chan et al.	
2001/0018721	A1 *	8/2001	McKenna et al.	710/126
2003/0005373	A1	1/2003	Michael	
2003/0115385	A1 *	6/2003	Adamane et al.	710/22
2003/0237032	A1 *	12/2003	Adler et al.	714/718
2004/0216018	A1 *	10/2004	Cheung	714/724
2005/0149693	A1 *	7/2005	Barry	712/34

(21) Appl. No.: **10/714,386**

(22) Filed: **Nov. 14, 2003**

(65) **Prior Publication Data**

US 2005/0107987 A1 May 19, 2005

(51) **Int. Cl.**  
**G06F 17/40** (2006.01)

(52) **U.S. Cl.** ..... **702/186**

(58) **Field of Classification Search** ..... 702/108,  
702/117-123, 186, 192; 714/9, 24, 42, 718,  
714/724; 717/105; 395/701, 705; 345/349;  
711/170; 707/1

See application file for complete search history.

(56) **References Cited**

U.S. PATENT DOCUMENTS

4,315,330	A *	2/1982	Brickman et al.	370/241
5,355,489	A *	10/1994	Bealkowski et al.	713/2

OTHER PUBLICATIONS

<http://www.merriam-webster.com/dictionary/operating%20system>, 1961.\*

GB Search Report for Application No. GB0425047.8 mailed on Feb. 24, 2005 (4 pages).

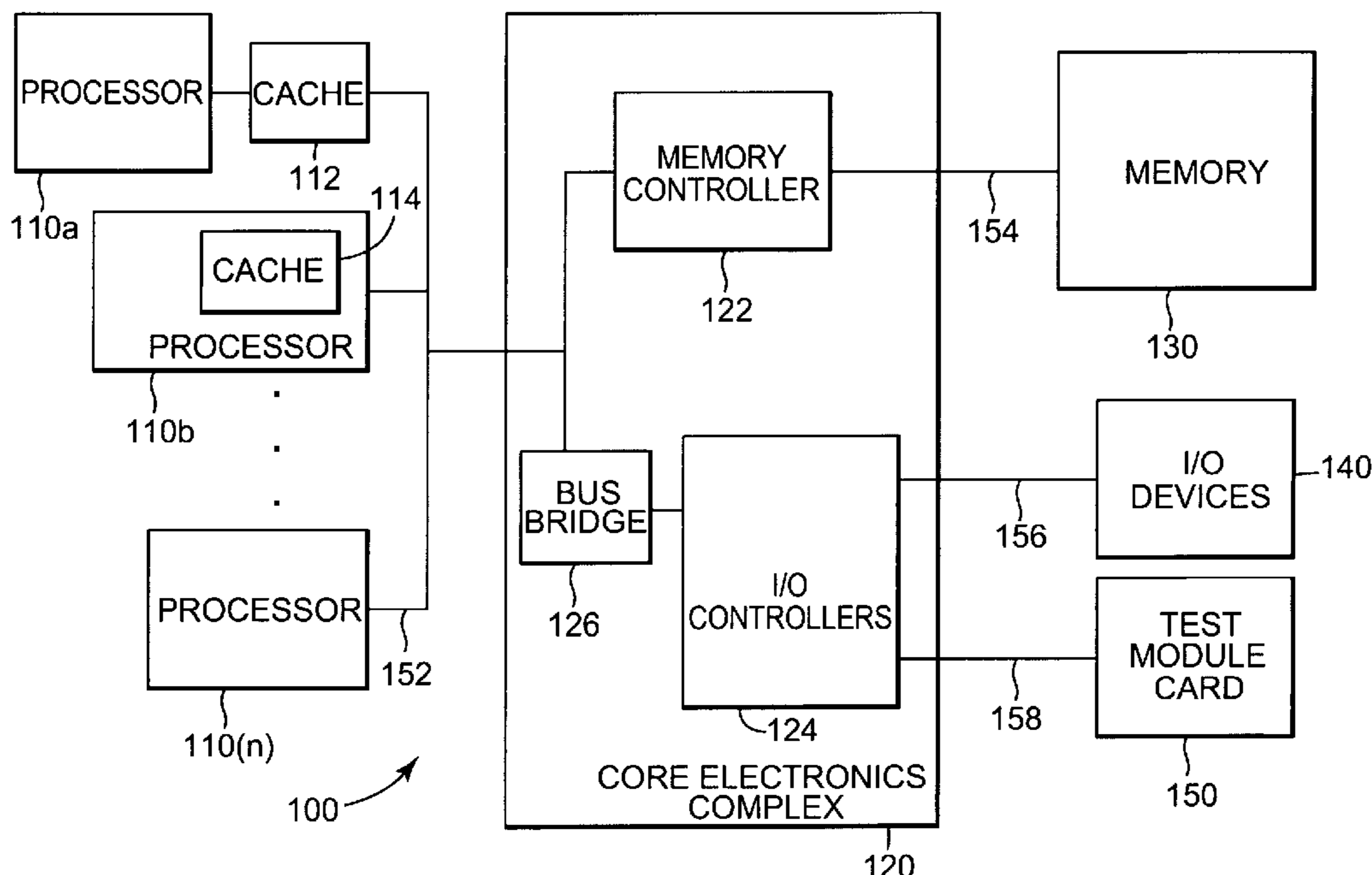
\* cited by examiner

Primary Examiner—Aditya Bhat

(57) **ABSTRACT**

A computer system that includes a processor, a memory controller coupled to the processor, a memory coupled to the memory controller, a first input/output (I/O) controller coupled to the memory controller, a first expansion slot coupled to the first I/O controller, and a test module card coupled to the first expansion slot wherein the test module card is configured to cause tests to be performed on the memory using direct memory access (DMA) is provided.

**20 Claims, 4 Drawing Sheets**



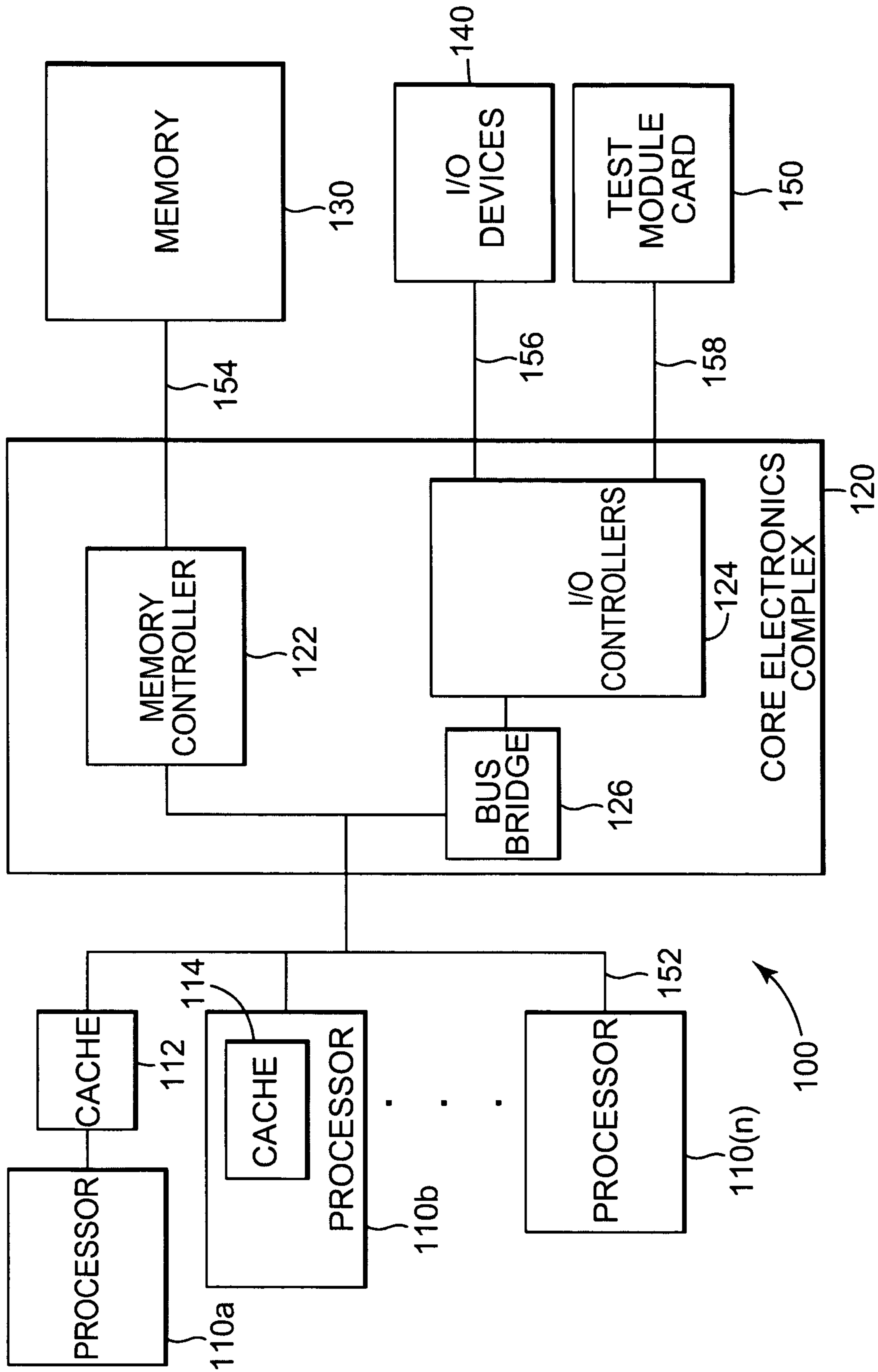


Fig. 1

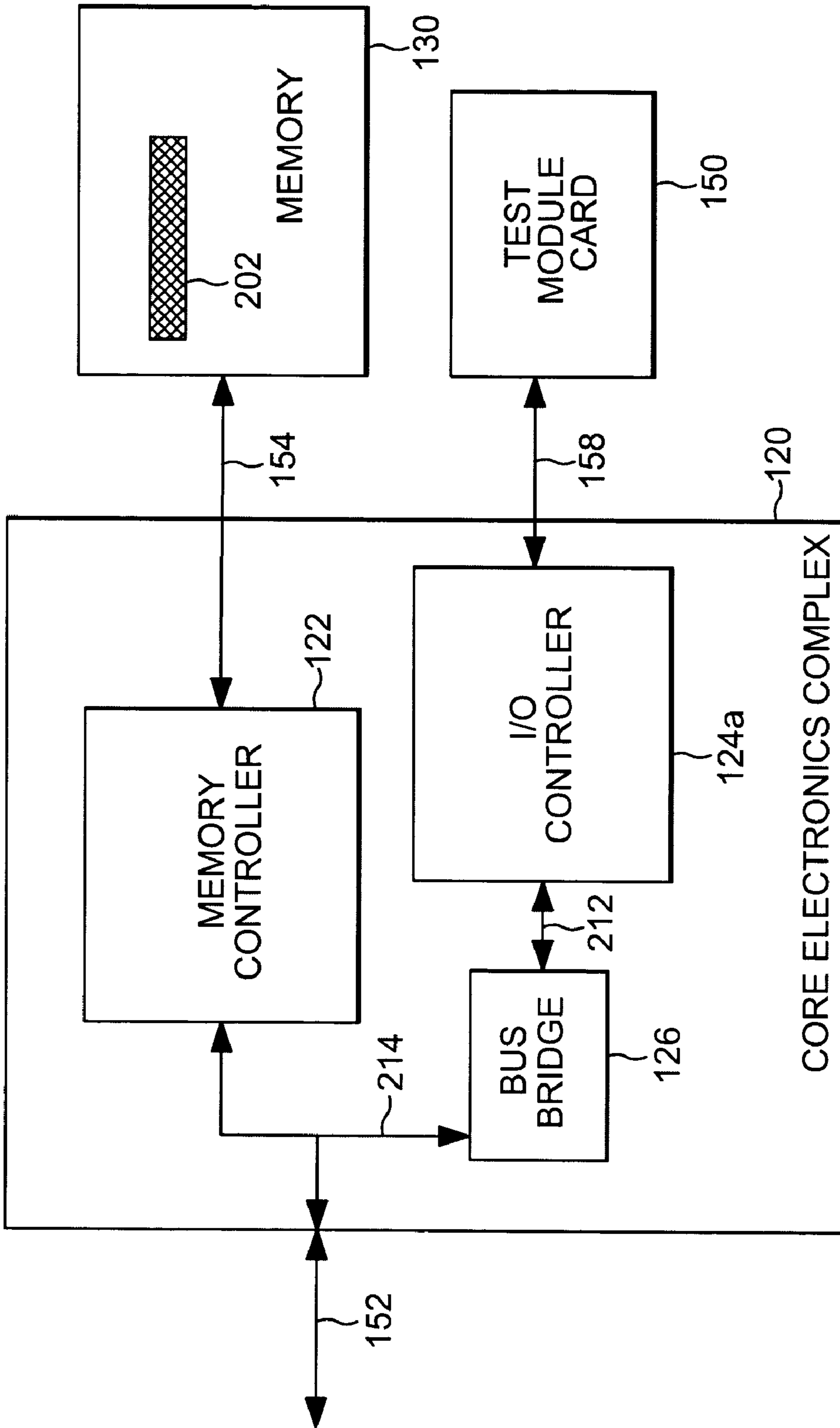


Fig. 2

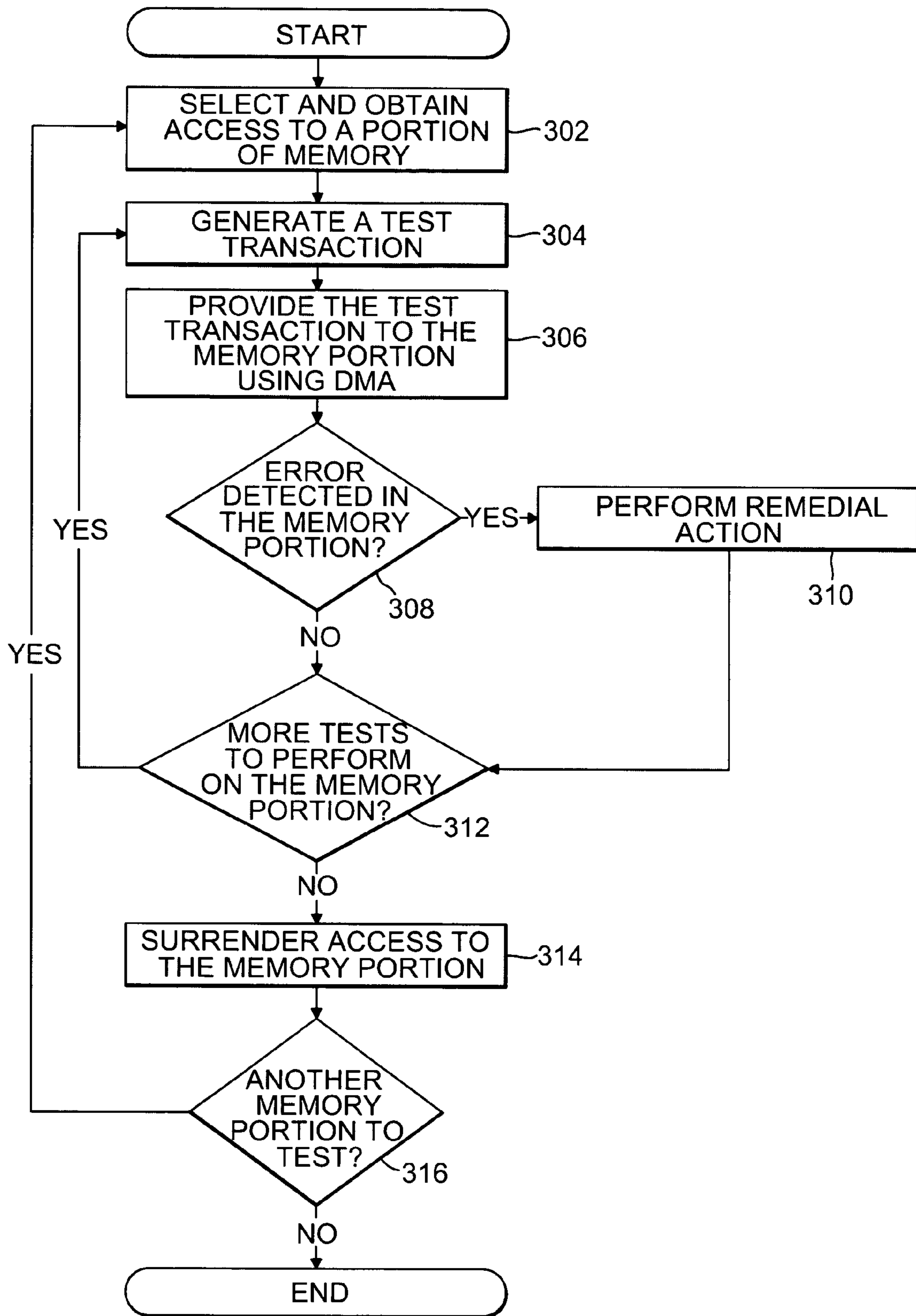


Fig. 3

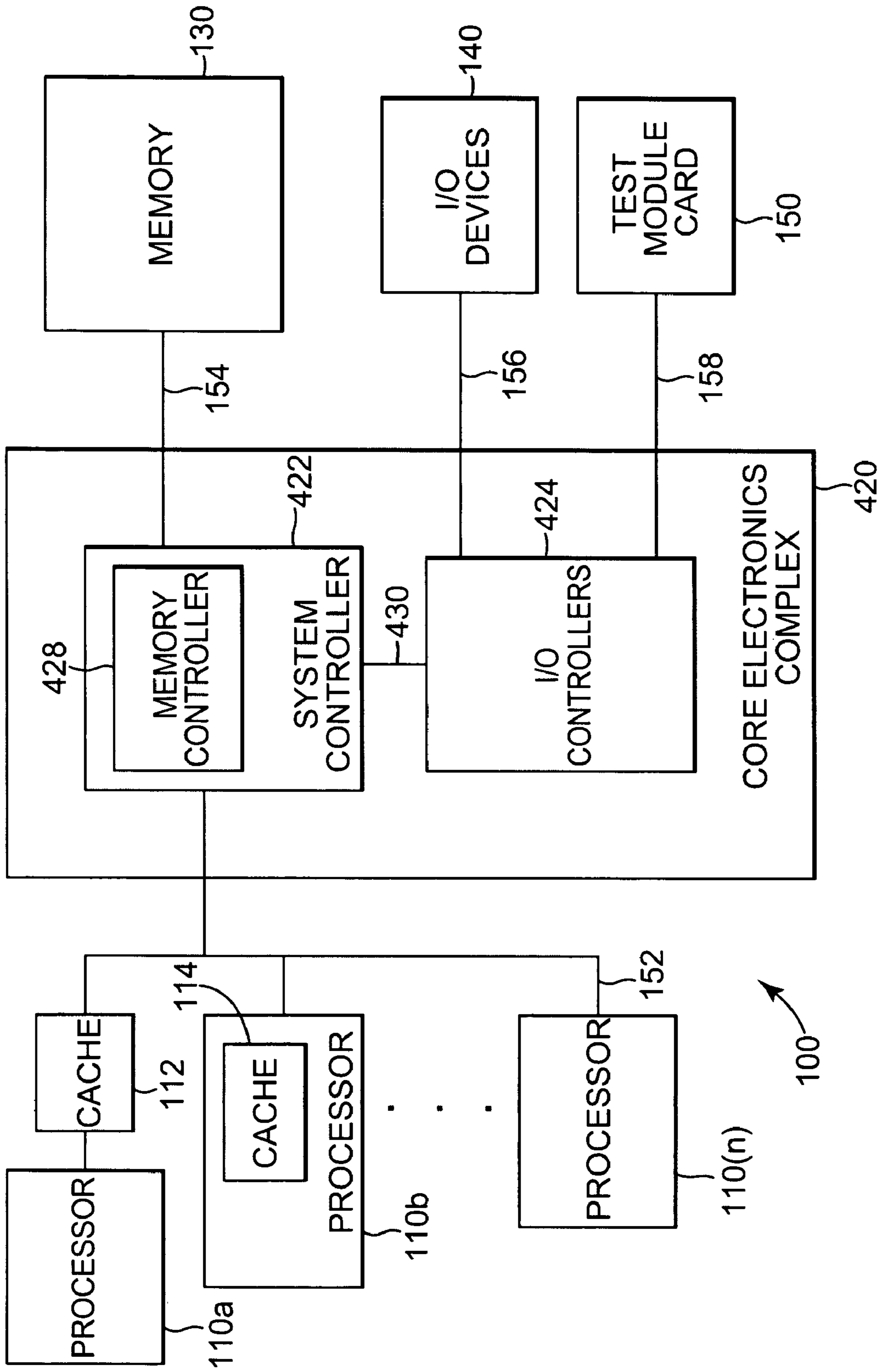


Fig. 4



## 1

**SYSTEM AND METHOD FOR TESTING A  
MEMORY WITH AN EXPANSION CARD  
USING DMA**

## BACKGROUND

Computer systems generally include one or more processors and a memory system. The memory system often includes multiple levels of memory devices that range from relatively fast and expensive memory to relatively slow and inexpensive memory. One of the first levels of a memory system is referred to as main memory usually comprises some form of Random Access Memory (RAM). In operation, a computer system loads an operating system and one or more applications into the main memory so that they may be executed by the processor(s).

Because the main memory contains the operating system and applications, it can be a critical component of the computer system. Failures that occur in the main memory can cause broader failures to occur in the system and possibly cause the system to crash. As a result, it is generally desirable to detect errors in the main memory before they cause failures.

Memory errors may be detected by writing known information to a memory and then reading the information back to determine whether it is correct. Some memory errors, however, may be pattern sensitive and may only appear in response to selected information patterns being written to the memory. Some diagnostic testing of a memory may occur in response to a computer system being turned on or reset. This type of testing, however, may not detect errors in computer systems that are left on and not reset for extended periods of time.

Although some memory devices include error correction features that work during operation of a computer system, these features typically detect errors only in response to a specific memory location being read. Because many areas of a memory may not be read with regularity, errors that occur in these areas may go undetected until an access to a faulty memory location takes place.

Accordingly, it would be desirable to be able to detect errors in all areas of a main memory of a computer system before the errors cause failures to occur during operation of the system.

## SUMMARY

According to one exemplary embodiment, a computer system is provided that includes a processor, a memory controller coupled to the processor, a memory coupled to the memory controller, a first input/output (I/O) controller coupled to the memory controller, a first expansion slot coupled to the first I/O controller, and a test module card coupled to the first expansion slot wherein the test module card is configured to cause tests to be performed on the memory using direct memory access (DMA) is provided.

## BRIEF DESCRIPTION OF THE DRAWINGS

FIG. 1 is a block diagram illustrating an embodiment of a computer system that includes a test module card configured to test a memory during operation using DMA.

FIG. 2 is a block diagram illustrating an embodiment of portions of the computer system shown in FIG. 1.

FIG. 3 is a flow chart illustrating an embodiment of a method for testing a memory during operation of a computer system using DMA.

## 2

FIG. 4 is a block diagram illustrating an alternative embodiment of a computer system that includes a test module card configured to test a memory during operation using DMA.

## DETAILED DESCRIPTION

In the following detailed description of the preferred embodiments, reference is made to the accompanying drawings which form a part hereof, and in which is shown by way of illustration specific embodiments in which the invention may be practiced. It is to be understood that other embodiments may be utilized and structural or logical changes may be made without departing from the scope of the present invention. The following detailed description, therefore, is not to be taken in a limiting sense, and the scope of the present invention is defined by the appended claims.

In one aspect of the present disclosure, a computer system includes a test module card coupled to an input/output (I/O) expansion slot. The test module card is configured to test a portion of the main memory of the computer system during operation of the computer system, i.e., subsequent to the operating system being booted and during execution of the operating system. The test module card provides test transactions to an I/O controller coupled to the I/O expansion slot which causes the tests to be performed on the main memory using direct memory access (DMA). The test module card may cause remedial action to be taken in response to errors being detected during the tests.

FIG. 1 is a block diagram illustrating an embodiment of a computer system **100** configured to test a memory during operation using direct memory access (DMA). Computer system **100** may be any type of computer system such as a handheld, desktop, notebook, mobile, workstation, or server computer. Computer system **100** includes processors **110a** through **110(n)**, a core electronics complex **120**, a memory **130**, a set of input/output (I/O) devices **140**, and a test module card **150**. Processors **110a** through **110(n)** are each coupled to core electronics complex **120** using a set of bus connections **152**. Bus connections **152** comprise a set of system busses. Core electronics complex **120** is coupled to memory **130** and I/O devices **140** using connections **154** and **156**, respectively. Core electronics complex **120** may also be referred to as a chipset.

Computer system **100** includes any number of processors **110** greater than or equal to one. As used herein, 'processor **110**' refers to any one of processors **110a** through **110(n)**, and 'processors **110**' refers to the set of processors **110a** through **110(n)**.

Processor **110a** is coupled to a cache **112**, and processor **110b** includes a cache **114**. Caches **112** and **114** may store any type of information such as instructions and data. Other processors **110** may include or be operable with any type or number of caches.

Computer system **100** also includes an operating system (not shown) that is executable by one or more of processors **110**. In response to being turned on or reset, one or more of processors **110** cause the operating system to be booted and executed. Processors **110** execute instructions from the operating system and other programs using memory **130**.

Core electronics complex **120** includes a memory controller **122**, a set of I/O controllers **124**, and a bus bridge **126**. Memory controller **122** is configured to store information into and read information from memory **130** in response to write and read transactions, respectively, from processors **110**, I/O devices **140**, and test module card **150**. Memory controller **122** may include hardware and/or software configured to



perform memory scrubbing or other error correction functions on memory 130 in response to reading information from memory 130.

I/O controllers 124 may include any type and number of controllers configured to manage one or more I/O devices 140 and test module card 150. Examples of I/O controllers 124 include IDE/ATA controllers, SATA controllers, PCI controllers, SCSI controllers, USB controllers, IEEE 1394 (Firewire) controllers, PCMCIA controllers, parallel port controllers, and serial port controllers. In one embodiment, I/O controllers 124 comprise multiple microchips that include an intermediate bus coupled to bus bridge 126, PCI controllers coupled to the intermediate bus, and SCSI, IDE and others controllers coupled to the PCI controllers.

Memory 130 comprises any type of memory managed by memory controller 122 such as RAM, SRAM, DRAM, SDRAM, and DDR SDRAM. In response to commands from system firmware (not shown) or the operating system, memory controller 130 may cause information to be loaded from an I/O device 140 such as a hard drive or a CD-ROM drive into memory 130.

I/O devices 140 may include any type and number of devices configured to communicate with computer system 100 using I/O controllers 124. Each I/O device 140 may be internal or external to computer system 100 and may couple to an expansion slot that is in turn coupled to an I/O controller 124. I/O devices 140 may include a network device (not shown) configured to allow computer system 100 to communicate with other computer systems and a storage device (not shown) configured to store information.

Test module card 150 is configured to cause tests to be performed on memory 130 during operation of computer system using DMA by generating test transactions and providing the test transactions to one of I/O controllers 124. The test transactions include read and write transactions configured to cause information to be read from and written to memory 130, respectively. The test transactions may include pattern tests or any other type or combination of read and write transactions configured to verify the operation and functionality of selected locations in memory 130.

Test module card 150 comprises an expansion card that is configured to plug into an expansion slot in a motherboard of computer system 100 or a connector coupled to the expansion slot and mounted in a chassis that houses computer system 100. The expansion slot couples to an I/O controller 124. Accordingly, test module card 150 may be internal or external to computer system 100. In embodiments where test module card 150 is external to computer system 100, test module card 150 may be housed in a chassis.

Computer system 100 includes multiple DMA channels that allow test module card 150 and I/O devices 140 to interact with memory 130. To use DMA, test module card 150 or an I/O device 140 sends a DMA transaction, such as a test transaction, to I/O controller 124. The DMA transaction goes from I/O controller 124 to bus bridge 126 then to memory controller 122 using a bus connection 152. Memory controller 122 causes the DMA transaction to be performed.

Test module card 150 may be tailored to operate in conjunction with memory controller 122 depending on the level of error correction functionality that is included in memory controller 122. For example, in embodiments where memory controller 122 implements memory scrubbing or other error correction mechanisms such as CRC codes, test module card 150 may be configured to primarily generate read transactions. In response to the read transactions, the memory scrubbing or error correction functions of memory controller 122 detect and, if possible, fix errors in memory 130. In embodiments

where memory controller 122 does not include memory scrubbing or other error correction mechanisms, test module card 150 may be configured to write patterns to memory 130 and read back the patterns from memory 130 to detect errors.

The operation of test module card 150 will now be described with reference to FIGS. 2 and 3. FIG. 2 is a block diagram illustrating an embodiment of portions of computer system 100, and FIG. 3 is a flow chart illustrating an embodiment of a method for testing memory 130 during operation of computer system 100 using DMA. The method is performed by test module card 150.

Test module card 150 selects and obtains access to a portion 202 of memory 130 during operation of computer system 100 as indicated in a block 302. Depending on the implementation, test module card 150 may obtain access to portion 202 by sending a request to the operating system, by sending a request to memory controller 122, by allocating an unused portion of memory 130, or by any other suitable method. In embodiments where memory controller 122 implements memory scrubbing or other error correction mechanisms, test module card 150 may omit the step of obtaining access to a portion of memory 130 before testing the portion.

Test module card 150 generates a test transaction as indicated in a block 304. The test transaction may be a read or write transaction that is configured to cause information to be read from or written to portion 202 in memory 130, respectively. Test module card 150 provides the test transaction to memory 130 using DMA as indicated in a block 306. In the embodiment of FIG. 2, test module card 150 accomplishes this by providing the test transaction to an I/O controller 124a using a connection 158. I/O controller 124a provides the transaction to bus bridge 126 using a connection 212. Bus bridge 126 provides the transaction to memory controller 122 using a bus connection 214. Bus connection 214 comprises one of the bus connections 152. Memory controller 122 causes the test transaction to be performed by reading information from or storing information to portion 202 in memory 130. In the case of a read transaction, memory controller 122 causes information to be read from portion 202 and provided to test module card 150 using connections 214, 212, and 158.

A determination is made by test module card 150 as to whether an error has been detected in portion 202 as indicated in a block 308. If test module card 150 detects an error in portion 202, then remedial action is performed as indicated in a block 310. Remedial action may include fixing errors, logging addresses of errors, and/or notifying the operating system, memory controller 122, and/or a system administrator of the errors.

If test module card 150 does not detect an error in portion 202, then a determination is made by test module card 150 as to whether there are more tests to perform on portion 202 as indicated in a block 312. If there are more tests to perform on portion 202, then the function in block 304 is repeated.

If there are no more tests to perform on portion 202, then test module card 150 surrenders access to portion 202 as indicated in a block 314. The function of block 314 may be omitted in embodiments where test module card 150 does not need to obtain access to portions of memory 130.

A determination is made by test module card 150 as to whether there is another portion in memory 130 to test as indicated in a block 316. If there is another portion in memory 130 to test, then the function of block 302 is repeated. If there is not another portion in memory 130 to test, then the method ends.

In embodiments where memory controller 122 is configured to perform memory scrubbing or other error correction functions in response to reading information from memory



130, errors may be detected and remedial action may be performed by memory controller 122 in response to the test transactions from test module card 150. The remedial action may include fixing and logging the errors.

In certain embodiments, computer system 100 includes software diagnostics that can communicate with test module card 150. In conjunction with the diagnostics, test module card 150 may perform targeted read, write, and pattern testing of memory 130. In response to obtaining access to a portion of memory 130, test module card 150 may test specific locations in memory 130 through any combination of reads, writes and pattern testing in response to communications from the diagnostics. The diagnostics may be included as part of the operating system or other applications or may be separate programs.

Test module card 150 may be designed to allow a system administrator to configure tests to memory 130 using the operating system or another suitable interface. The operating system or interface may allow the administrator to select, for example, the time and frequency of tests, the particular portions of memory to test (e.g. frequently used portions), and the actions to be taken in the event of failures.

Test module card 150 may comprise any suitable combination of hardware and software components.

FIG. 4 is a block diagram illustrating an alternative embodiment of computer system 100 configured to test memory 130 during operation using DMA. In the embodiment shown in FIG. 4, core electronics complex 420 replaces core electronics complex 120. Core electronics complex 420 includes a system controller 422 and I/O controllers 424. System controller 422 includes a memory controller 428.

Test module card 150 operates in substantially the same way in the embodiment of FIG. 4 as it did in the embodiments of FIGS. 1, 2, and 3. In the embodiment of FIG. 4, DMA transactions are performed using one of a set of connections 430 between an I/O controller 424 and system controller 422 instead of using a bus connection 152. Accordingly, the test transactions of test module card 150 go from test module card 150 to an I/O controller 424 to system controller 422 using a connection 430. Memory controller 428 processes the test transactions and, in the case of read transactions, provides information associated with the read transactions back to the I/O controller 424 using connection 430.

As illustrated by FIG. 4, test module card 150 is configured to operate with different arrangements of core electronic components or chipsets. Accordingly, test module card 150 is also configured to operate with other core electronic components or chipsets not illustrated herein.

The above embodiments may provide advantages over systems that do not use DMA to detect memory errors. For example, a computer system may more easily correct memory errors detected using DMA than with other types of memory error detection. In addition, DMA accesses may require less processor overhead than operating system-level processes. Further, operating system-level processes may have recovery problems associated with the operating system.

Although specific embodiments have been illustrated and described herein, it will be appreciated by those of ordinary skill in the art that a variety of alternate and/or equivalent implementations may be substituted for the specific embodiments shown and described without departing from the scope of the present invention. This application is intended to cover any adaptations or variations of the specific embodiments discussed herein. Therefore, it is intended that this invention be limited only by the claims and the equivalents thereof.

What is claimed is:

1. A computer system comprising:

a processor configured to execute an operating system;  
a memory controller coupled to the processor;  
a memory coupled to the memory controller;  
a first input/output (I/O) controller coupled to the memory controller;  
a first expansion slot coupled to the first I/O controller; and  
a test module card directly coupled to the first expansion slot;

wherein the test module card is configured to obtain access to a portion of the memory from the operating system, and wherein the test module card is configured to cause tests to be performed on the portion of the memory using direct memory access (DMA) subsequent to obtaining access to the portion of the memory.

2. The computer system of claim 1 wherein the processor is configured to cause the operating system to be booted, and wherein the test module card is configured to cause the tests to be performed on the portion of the memory subsequent to the operating system being booted.

3. The computer system of claim 1 wherein the test module card is configured to cause the tests to be performed on the portion of the memory during execution of the operating system.

4. The computer system of claim 1 further comprising:  
a second I/O controller coupled to the memory controller;  
a second expansion slot coupled to the second I/O controller; and

an I/O device coupled to the second expansion slot.

5. The computer system of claim 1 wherein the test module card is configured to cause tests to be performed on the memory by providing read and write transactions to the first I/O controller.

6. The computer system of claim 5 wherein the read and write transactions comprise DMA transactions.

7. The computer system of claim 1 further comprising:  
a bus bridge coupled to the processor and the first I/O controller.

8. The computer system of claim 1 further comprising:  
a system controller that comprises the memory controller.

9. A method comprising:

obtaining access to a portion of a memory of a computer system from an operating system during operation of a computer system;

generating a test transaction in a test module card directly coupled to an expansion slot of the computer system; and

providing the test transaction to the portion using direct memory access (DMA) to cause information to be read from or stored into the portion subsequent to obtaining access to the portion of the memory.

10. The method of claim 9 further comprising:

detecting an error that occurs in response to the test transaction; and

performing a remedial action associated with the portion in response to detecting the error.

11. The method of claim 9 further comprising:

providing the test transaction from the test module to an I/O controller coupled to the expansion slot;

providing the test transaction from the I/O controller to a bus bridge;

providing the test transaction from the bus bridge to a system bus;

providing the test transaction from the system bus to a memory controller; and



7

providing the test transaction from the memory controller to the portion.

**12.** The method of claim **11** further comprising:

storing information in the memory in response to the test transaction being a write transaction. 5

**13.** The method of claim **11** further comprising:

in response to the test transaction being a read transaction:

providing information associated with the test transaction from the portion to the memory controller;

providing the information from the memory controller to the system bus; 10

providing the information from the system bus to the bus bridge;

providing the information from the bus bridge to the I/O controller; and 15

providing the information from the I/O controller to the test module.

**14.** The method of claim **9** further comprising:

providing the test transaction from the test module to an I/O controller coupled to the expansion slot; 20

providing the test transaction from the I/O controller to a system controller;

providing the test transaction from the system controller to a memory controller; and

providing the test transaction from the memory controller to the portion. 25

**15.** A computer system comprising:

a processor;

a memory controller coupled to the processor and configured to perform error correction;

a memory coupled to the memory controller;

an input/output (I/O) controller coupled to the memory controller; 30

8

an expansion slot coupled to the I/O controller; and a test module card directly coupled to the expansion slot;

wherein the test module card is configured to obtain access

to a portion of the memory from an operating system,

and wherein the test module card is configured to cause

tests to be performed on the portion of the memory by

providing read transactions associated with the memory

to the I/O controller subsequent to obtaining access to

the portion of the memory.

**16.** The computer system of claim **15** further comprising:

the operating system;

wherein the processor is configured to cause the operating

system to be booted, and wherein the test module is

configured to cause the tests to be performed on the

memory using DMA subsequent to the operating system

being booted.

**17.** The computer system of claim **15** further comprising:

the operating system;

wherein the processor is configured to cause the operating

system to be executed, and wherein the test module is

configured to cause the tests to be performed on the

memory using DMA during execution of the operating

system.

**18.** The computer system of claim **15** wherein the I/O

controller provides the read transactions to a system bus.

**19.** The computer system of claim **15** wherein the test

module card is configured to cause tests to be performed on

the memory using direct memory access (DMA).

**20.** The computer system of claim **15** wherein the read

transactions comprise direct memory access (DMA) transac-

tions.

\* \* \* \* \*